

## **N5 Assignment**

**Website name: </> (otherwise known as Brackets)**

**Student number: 2721431**

For the CSCU9N5 website we were required to design a web-based multimedia application with the topic of it being any one of six choices. The one that I chose was to make an interactive instructional course as I felt that would allow me to utilise the strengths that I already had in web design and the ideas for it came pouring into my head as soon as I had looked at said topic.

The plan was for the website to be used by people beginning or interested in starting programming and computing as it would lay a focus on the very basics of three distinct, well-known programming languages in Java, C++ and Python.

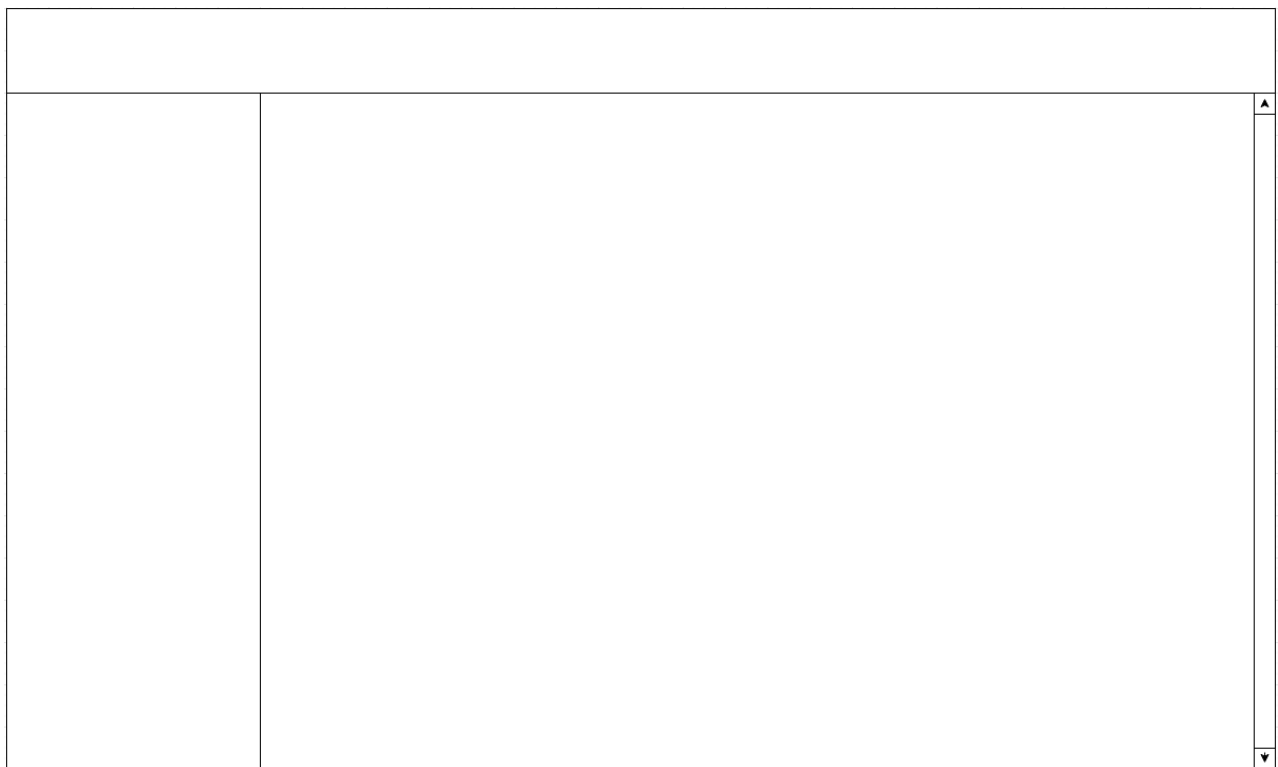
Someone like Markus Black, age 17, who goes to a public school where he's averagely popular, but he doesn't really know what he wants to do going into his future. He's heard about good job opportunities in the field of Computing and Programming so even if he doesn't go to college or University, he wants to learn some programming to help him have a chance at getting a well-paying job in the future and he has found a few project ideas that he could try working on, however he has no idea how to start them out. He does well in his subjects and he is a fast learner. One day he would be browsing the web looking for anything that would help him understand how to begin with programming a project of his that requires him to make a program that would record results and running times during his school sports day races and compare them to an inbuilt system that would see if the racers will have qualified for the next round or not. Being able to get this project working would allow him to get on the good side of his sports teachers, almost guaranteeing a good result from them on his report card. He would come across this website and see the basics laid out for him to learn from and understand how to move forward with his current project and all future projects.

Someone else who could use a website like this is John Tubman, age 20, who recently started university doing a course in computer science, but he has been struggling with his practical work. He is very interested in the subject when it comes to the theory but when it comes to putting what he has learned into practice he always seems to fall short and it has been demoralising for him so far. He is a very motivated student however and he is trying whatever he can to help himself get through the programming sections of his assignments. He lives on his own and takes relatively good care of himself and in his spare time he enjoys gaming, which is what initially got him into computer science, hoping that one day he would be able to implement what he has learned in programming into making his own game. Like Markus, he would be browsing the web, struggling on his most recent practical that requires him to use a loop to count how large an array containing student names is. He barely managed to get the if...else statements working and this practical has truly stumped him. However, thanks to this

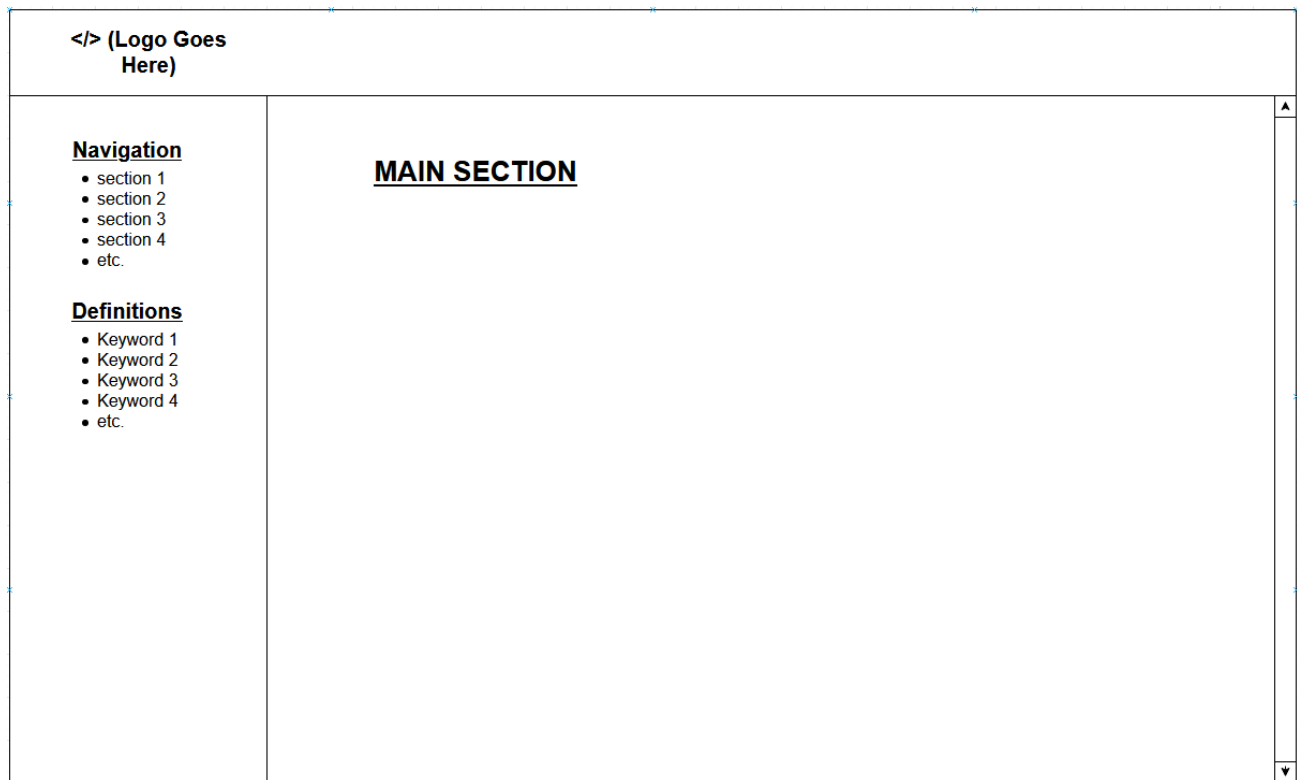
website containing the instructional course he can get a true grasp of the basics of programming and get the help he needs to complete his practical.

Generally, the people who would use this website want to learn a new skill that would be extremely beneficial for them in their future therefore, the website is for anybody wanting to start to code so it is made to be accessible by whoever wants to use it. However, according to a survey made by Hackerrank in 2018(<https://research.hackerrank.com/developer-skills/2018/>) most people start learning how to code around the ages of 16 to 20 and this website is aimed mostly at those people and older and I would advertise and market it towards users of that age.

For the design then I decided to go with a basic, clean structure for the skeleton of the website so as not to overwhelm anyone coming across it for the first time:



As you can see, there is a bar at the top which is the header which is where the logo would go. The left column would be used for navigating around the website and any extra information required. The main section would in the area to the right.



This way all the information is condensed in the main section whilst the rest of the webpage can look uncluttered so as not to throw too much information at someone using the website for the first time. The logo was chosen to go into the top left corner due to how familiar people would be with any logo being in that top-left hand corner. Huge company and corporation sites like the BBC, Microsoft Outlook, Amazon, Netflix and John Lewis and Partners all have their logos in the top left-hand corner allowing for people to familiarise themselves with a logo being there now and always. However, I chose for it to be on its own in the header to avoid clutter and to keep that clean website design. That familiar feeling is also a big reason why I decided to put the navigation and definition section on the left of the website and the main section to the right. However, that is not the only reason. There was a study done at Cleveland State University that showed that people reacted negatively when watching footage featuring movement from the right side of the screen to the left side of screen whilst they reacted positively to the opposite (<https://nofilmschool.com/2016/02/left-or-right-why-characters-lateral-movement-screen-matters-film>). Now although the study was conducted with film and media in mind its findings can be very easily applied to designing websites as well, where looking at content on the right side of the screen exhibits a more positive response than looking at content on the left side. It is for that reason also I put my main section onto the middle to right side of the website.

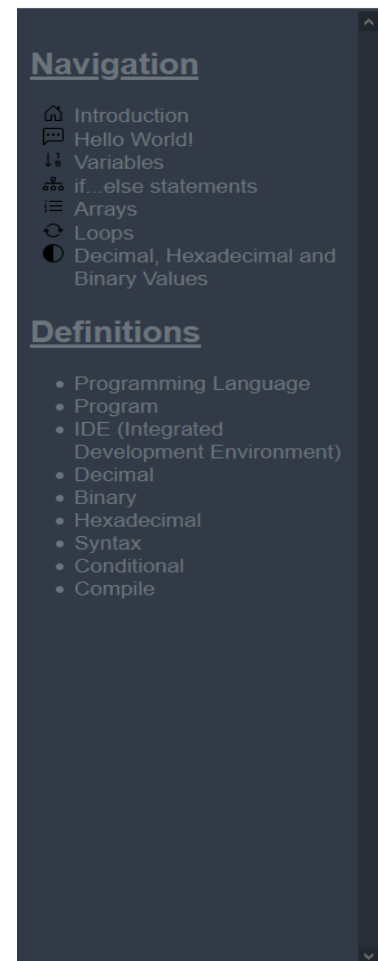
So now why is the navigation bar on the left other than to keep the website design clean if people react negatively from looking from left to right? This is because I have found that, through myself and others that I have used to test the website, when using a mouse (a touchpad didn't show any noticeable differences), it feels more natural to move a cursor

horizontally across a screen as opposed to moving it vertically due to the positioning your wrist takes when moving the mouse around. For that reason, I placed that column to the left side of the screen to allow for the navigation of the website to feel as natural as possible.

The right section was also made to be scrollable independently of the rest of the website so that the section for navigating the websites and the area where you can look up any definitions quickly without disturbing yourself from reading is always accessible.

For the left section the colour that was chosen is on the darker side of the colour spectrum to add a contrast with the rest of the website, which was white and off-white, to make it more noticeable as the user would be looking at the content of the website. The letters are of a darker colour to make the reading of them easier on the eyes as extended periods of brighter colours on computer screens can make our eyes sore so the darker colours are a welcome break from looking at the rest of the screen, even if it is for a moment. The left section also uses metaphors in the positions where bullet points would usually be to help the user understand what each area on the website is about. Introduction uses a symbol that looks like a house, where the beginning of the webpage is, home. Hello World! Uses a speech bubble to represent the first output or in this case voice of our program. Variables has an icon showing an arrow pointing from 1 to 9 showing the user that that's what variables is all about, numbers. If...else statements uses a tree diagram representing the different choices and paths taken when writing a conditional statement. Arrays is represented by a symbol of a list as that is what an arrays most basic function is. Loops has two arrows looping around themselves forming a circle showing the repetitive nature of a loop and finally, decimal, hexadecimal and binary values uses a circle with half black and the other half white, being a metaphor for how binary numbers are either represented using 1's or 0's or true or false or yes or no. These bullet points in navigation were placed in the same order that they show up in the main section on the right so that the user can more easily understand where they are on the website with just a glance.

The right section was not made to have a dark background however since, although darker colours are easier on the eyes, it can make sentences and paragraphs difficult to read. Now to still allow for care towards the eyes, the sections that contain content are not white. This is to distinguish those sections on the website instantly when glancing at the page and making them slightly more readable without sacrificing visibility of the words and letters.



## So What is Code?

When you learn to code you can make things happen on your computer.

You can make anything you want with code. It could be a game, some pictures or a film.

Computer code is a set of rules or instructions. It is made up of words and numbers and when you put them in the right order it will tell your computer what you want it to do. You can program lots of things with code. (Original Source: <https://www.bbc.co.uk/bitesize/topics/z3tbwmn/articles/zykx6sg>)



```
for i in people.data.users:
    response = client.api.statuses.user_timeline.get(screen_name=i.screen_name)
    print('Got', len(response.data), 'tweets from', i.screen_name)
    if len(response.data) > 0:
        tdata = response.data[0]['created_at']
        tdate2 = datetime.strptime(tdate, '%a %b %d %H:%M:%S +0000 %Y')
        today = datetime.now()
        howling = (today - tdate2).days
        if howling >= 14:
            print(i.screen_name, 'has tweeted in the past', daywindow,
                  'total tweets =', len(response.data))
            for j in response.data:
                for k in j.entities.urls:
                    newurl = k.expanded_url
                    urlset.add((newurl, j.user.screen_name))
            # Add
            print(i.screen_name, 'has not tweeted in the past', daywindow)
```

## What We Do

Here at </>, otherwise known as Brackets, we strive teach the world about coding one course at a time. We have basic courses all the way up to the most advanced and abstract code you may think of as well as advanced computational thinking.

We have other courses which touch upon Concurrency and Distribution of Operating Systems as well as Parallelism all the way up to basic Neural Networks.

On this course however we're just going to focus on the raw basics of programming languages that you may or may not have heard of. Three to be exact, Java, C++ and Python. It is using these three programming languages that we're going to move forward with our multiple courses one step at a time.

## Covering Binary and Hexadecimal

In the basics, as well as basic programming practices we're going to cover binary and hexadecimal, which are 2 of the most basic topics in Computer Science.

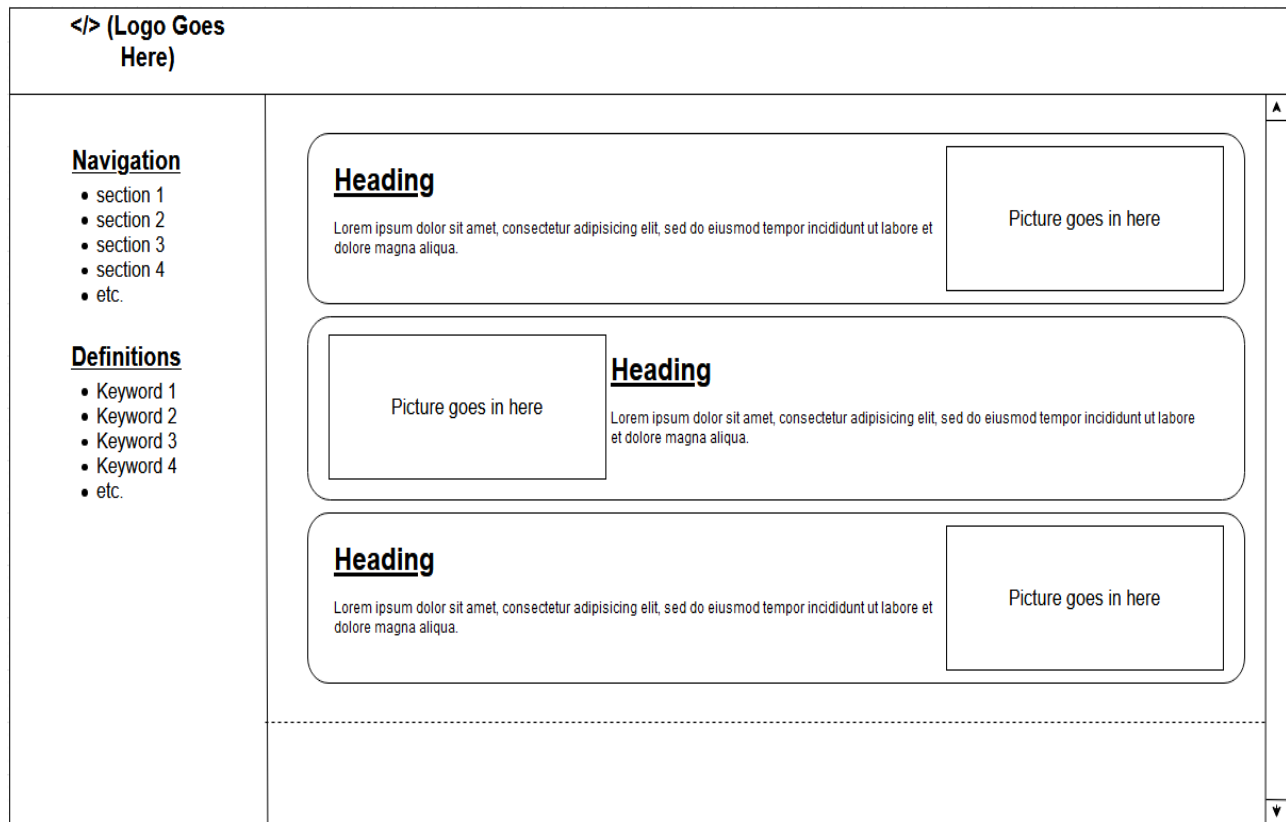
The reasoning behind learning binary is to understand how computers think at the bottom most level, even if the majority of programmers or even hardware engineers don't even deal directly with binary. That is where Hexadecimals come in. They are more readable than binary as they more closely resemble our own decimal(base-10) counting system it's therefore easier to see at a glance how big a number is in Hexadecimal as opposed to Binary. The other reason is touched upon further on in the Decimal, Hexadecimal and Binary values section of this webpage. (Original Source: <https://www.computerhope.com/jargon/h/hex.htm>)



## The "Hello World!" Function

The first and third pictures above were chosen to be added to the website to add some different colours to it other than the usual grey, white and black and to make it more pleasing to look at, especially since these are on the front page so they're the first thing you see when accessing the website. They were specifically chosen to be more blue in colour as multiple studies have shown that blue seems to have a more calming effect on people, as opposed to red which promotes aggression. This works out perfectly for people that are just starting out programming and might be a little panicked or nervous as programming can be quite overwhelming therefore, they can come into this website and their brains, when recognising the colour blue, will associate it with calmness and it can have a naturally relaxing effect. The second picture was chosen to show what some actual code would look like, albeit in an IDE using dark mode, and it was chosen to be dark mode for the same reasoning as in the left column of the website, to add some contrast and be easier to look at.

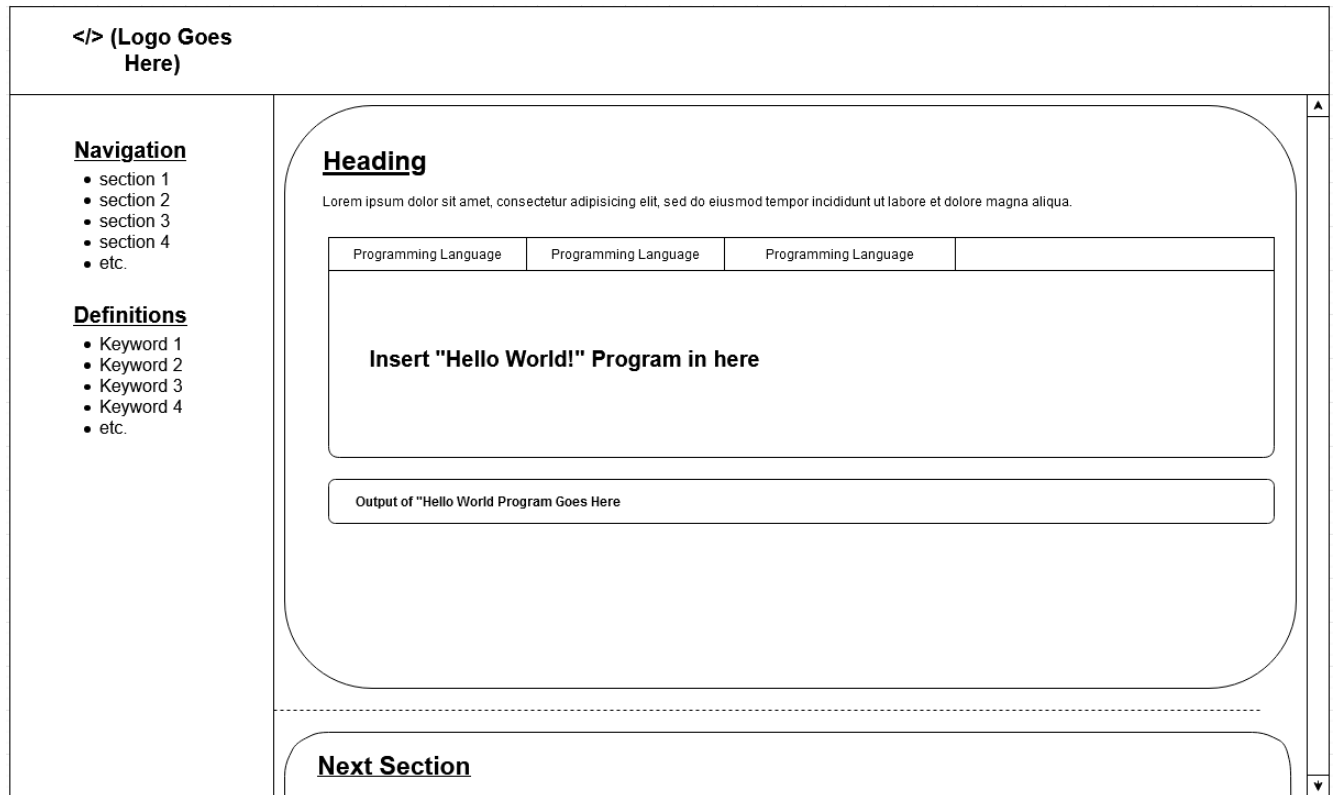
The Introductory section follows this format:



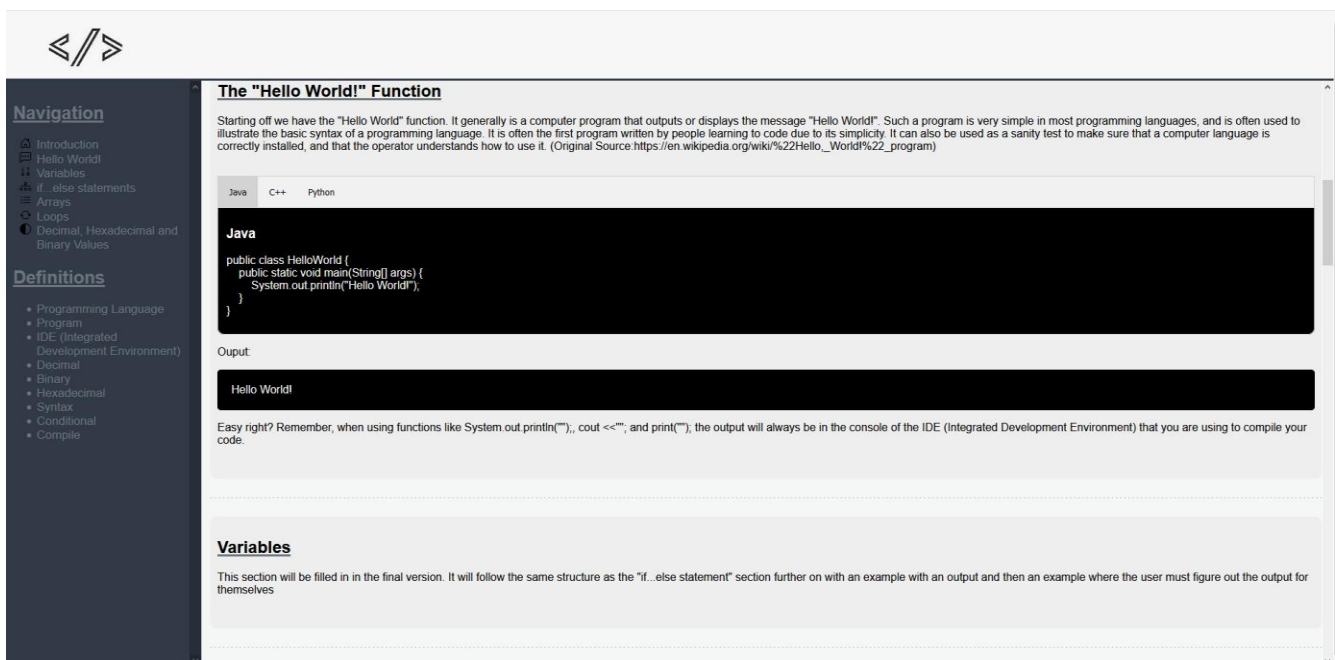
As you can see the different pieces of content are contained in their own sections. This is to promote engagement from the user as the structure of how the paragraphs are laid out allow for them to easily follow what is written and the changing of section works in the same way as changing a paragraph. It allows the brain to take a breather before reading a different section as opposed to where there would be a wall of text, overwhelming the reader. The pictures are chosen to be relevant to the topic in the section that they're contained in.

After the introductory section is done you will notice there is a dashed line creating a divide across the main section of the website. This is because I have chosen to develop the website all on one webpage in order to make navigation as easy and seamless as possible. Through asking around multiple people, I have found that most users prefer it when, if you click a link, it sends you to somewhere already existing on the webpage as that makes navigating the website much, much faster. If the link was to send you to a different webpage it is dependent on your internet and on your computer how quickly you get to that webpage. Therefore, having everything already loaded onto one allows for smooth and easy transitioning throughout the website without having to rely entirely on external factors affecting your navigation through it. The dashed line is there to show when one big section ends and the next begins. In the case of the Introductory section, the part after it would be the "Hello World" section. This, like every other section on this website, is easily accessible by scrolling or by quickly jumping to it using the link in the Navigation section on the left side.

The "Hello World" section follows this format:



And completed looks like this:



Clicking on the Hello World! link in the Navigation section will send you to the Hello World! Part in the main section and it works that way with each link in the navigation section sending you to the corresponding area in the main section.

The Hello World part, like all the other programming parts, was designed to give the user an explanation of what the function does exactly and why it is used in order to not throw the user into the code blind. This way there is some understanding behind why they are learning the function.

There are three tabs representing the three programming languages that the person using the website can choose to learn. They're placed on the left side of the main section since people learn to read moving from left to right so when the user sees the programming area their instinct is to look at it from the left side first, seeing the tabs. They are designed to look clean and sleek with a white background when unclicked for readability and a grey background when clicked for noticeability.

The area which contains the code is black to add a contrast to the mostly white main section and to be easy on the eyes again, giving them a rest from all the brightness of the main page. The code is white to make it easily readable. The corners of this area as well as the sections which contain the content are all rounded to seem friendlier to the user as opposed to sharper corners as the brain naturally associates sharpness with danger and roundness with something harmless. It is also easier for our eyes to follow a rounded corner, noticing the sections more easily, as the path that the corner is taking is more obvious (<https://uxmovement.com/thinking/why-rounded-corners-are-easier-on-the-eyes/>). The output area follows the same logic as the programming area.

The if...else statement section follows a very similar structure. It has the same explanation area at the beginning, the same programming area and tabs and the same output area. Where it differs is that it allows for some interactivity with the user in which they must read through an if...else function and try to figure out the output for themselves.

The format looking a little like this:



</> (Logo Goes Here)

Navigation

- section 1
- section 2
- section 3
- section 4
- etc.

Definitions

- Keyword 1
- Keyword 2
- Keyword 3
- Keyword 4
- etc.

Heading

Programming Language

Programming Language

Programming Language

Insert Program in here

Output of Program Goes Here

Programming Language

Programming Language

Programming Language

Insert Try It Yourself Program in here

Possible Output 1

Possible Output 2

Possible Output 3

Possible Output 4

Correct or Incorrect

Underneath the Insert program in here section right before the output there will be a paragraph going through the function line by line to ensure the user of the website can fully understand what exactly is going on.

The styling of the try it yourself section will be the same as the Insert Program in here section, but the difference is that underneath the code area there will be four buttons with four possible outputs for the try it yourself section. These buttons have been stylised to turn green, a colour that's often associated with getting things right, if the output being clicked is the correct one and red, a colour that's often associated with getting things wrong, if the output being clicked is the wrong one. So, the buttons are all going to start off as the same white colour for readability and change when clicked. The reason green and red were chosen is because of familiarity with the colours and what they generally represent in the context of getting things right and wrong.

Underneath the buttons, after they have been clicked, an output will show up saying if they are correct or incorrect in the respective font colour of green if correct and red if incorrect for easy,

quick understanding of whether the user got the button press wrong or right. The word Incorrect also, was chosen over the word wrong in the case of the word that's being outputted because wrong is a harsher and more negative sounding word than incorrect. When people first start trying to learn coding, you want to try avoiding things that would demoralise them which is why I chose to go the more "positive" sounding word.

Again, all the content for the if...else statement section is contained in a lightly coloured box with rounded edges.

The rest of the programming sections will have structures very similar if not the same as the if...else statement section which was why they were not included in the prototype and merely mentioned how they would be completed in the final version since the template for how they will look is already there.

The final section on the website is the decimal, binary and hexadecimal section, where the most basic part of computer science is contained. I thought to add this section as it is always the first thing anyone ever learns when entering computing and, therefore, to possibly garner the interest of the person reading about it.

The format of the decimal, binary and hex section:

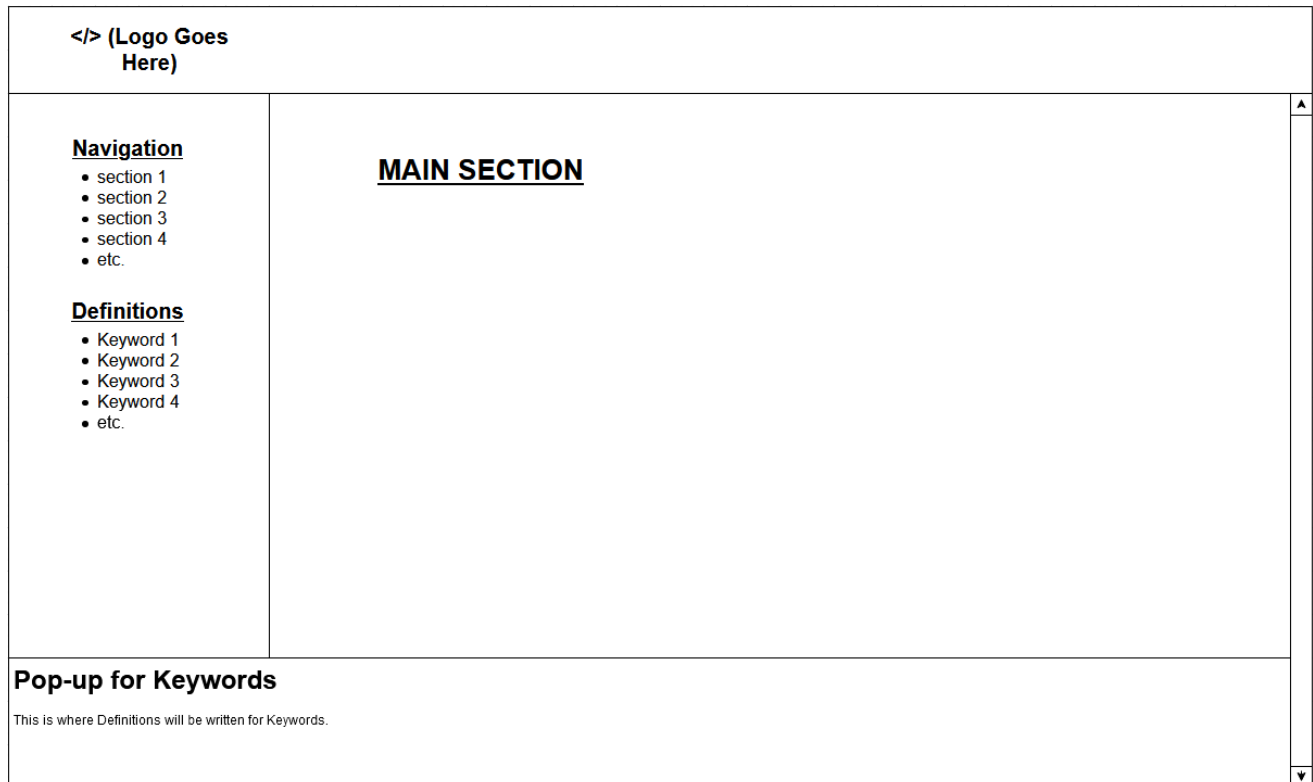
The wireframe shows a website layout with the following components:

- Header:** A box containing the text "</> (Logo Goes Here)".
- Left Sidebar:**
  - Navigation:** A list of links: section 1, section 2, section 3, section 4, etc.
  - Definitions:** A list of keywords: Keyword 1, Keyword 2, Keyword 3, Keyword 4, etc.
- Main Content Area:**
  - Heading:** A section titled "Heading" with a placeholder text "Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua."
  - Diagram:** A box titled "Diagram of how to convert Decimal into Binary" with a placeholder text "Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua."
  - Decimal to Binary to Hexadecimal Converter:** A box containing three input/output fields:
    - Decimal:** A box labeled "Enter Decimal Number Here".
    - Binary:** A box labeled "Binary Number Outputs Here".
    - Hexadecimal:** A box labeled "Hexadecimal Number Outputs Here".

Now as this is the most theoretical section there is a lot more reading involved as this could possibly be the first interaction the user has ever had with binary and hexadecimal numbers where they could be in a position to understand what is being talked about.

The general structure of the content is similar to the coding sections with the header being in the same place and explanation of what the topic is about being right below it. Diagrams are included to help the person reading to properly understand how binary and hexadecimal work.

Although the decimal, binary and hexadecimal values part is the last section visible on the webpage, there is one last part that is usually hidden unless the correct link is clicked. The definitions section on the left side of the screen contains keywords that are useful to know if the user does not already know them, almost like a mini dictionary. When clicked, a pop-up will come up at the bottom of the screen like this:



Having the definition of a keyword the user is stuck on come up like this allows for said user to continue reading the part they were going through whilst also seeing the definition of the word they were stuck on therefore, not losing track of where they were. To get rid of the pop-up they either can click anywhere on the page that's not part of the pop-up like most people will instinctively do or click in the bottom right corner of the screen a button that says 'ok' showing that they've understood what the keyword means. Both methods have been implemented to allow the user to quickly access a definition and just as quickly, get rid of it as efficiently as possible. It also transitions onto the screen with a smooth animation so as not to surprise the user who clicked one of the keywords on the left.

The interactive part of this section is a functioning decimal to binary to hexadecimal converter which will allow the user to input a decimal and output its respective binary and hexadecimal numbers. This should allow the person trying to learn from the course to test out their numbers so they can truly grasp an understanding of how the conversion from decimal to binary to hex works.

All things clickable also have a cursor change when you mouse over them, changing into a closed hand with its index finger and thumb outstretched. This was chosen because through the testing most people were familiar with this symbol being the one used whenever something is clickable and as such, they would know that they would be able to click it.

Everything that was built in the prototype website was built because anything that is added on in the final version would follow the same structure as the already existing parts. That's the criteria I used to build it and if there was anything new and unique that was going to be part of the website, I would add it in.

All of this should cover some of the important basics of programming to help people entering the field for the first time or struggling to start, learn these basics and improve on them in the future.

My main focus when designing the website was to make it easy to learn from and build upon whilst also, most importantly to myself and to hopefully the people I designed it for, keeping a clean, simple look because I didn't want the website to treat the users of it like children but instead like professionals, trying to learn a skill that is a huge help when looking for a future career.

The testing was first carried out by myself, to ensure that all links worked, all buttons clicked, resizing wasn't an issue and that any functionality wasn't unintentionally broken or faulty. Then I allowed the website to be tested by two people who also do computer science (not at Stirling though) to get a fresh new perspective on anything to improve on the website or anything in the code that looks like it doesn't make any sense. This allowed for a better source of feedback than for example asking someone in the same module as me as they would have had their own ideas in their head for their own website whilst I wanted someone who had no prior knowledge of the project to take a look at it and give their first impressions. Afterwards, the website was looked at by two more people one of which was around the initial target demographic of 16 to 20 years old and the other being someone who had only recently started to learn how to code was struggling with it. I took their feedback into consideration and implemented the improvements that they suggested into the website, one of those being a Contact Us button for any future feedback to be submitted by anonymous users of the website. This was implemented into the top-right corner of the website in the header as it was an area that had not much in it so a button there would stand out easily to anyone taking a glance at the webpage. It was also designed to look clickable and make a clicking animation when pressed upon.

Sources for all of all media, external CSS and JavaScript libraries:

- <https://www.jqueryscript.net/other/decimal-binary-hex-octal-converter.html> -> This site was used to source the CSS and JavaScript for the decimal to binary to hex converter. The JQuery wasn't working initially so I changed a few things here and there

to get it to work and I altered some parts of the CSS and applied it to the box with rounded corners that holds most of the content on the website.

- <https://icons.getbootstrap.com/> - > I downloaded all the icons from Bootstrap and then picked and chose which ones I liked best as replacements for the bullet points.
- <https://demo.mobiscroll.com/> -> Mobiscroll is a JQuery and CSS library I acquired from when I was working on web design over the summer. It was used to get the pop-up for the definitions working.
- <https://jquery.com/> -> JQuery allowed for all these other libraries to work properly and to run Ajax calls.
- <https://img-cdn.tnwcndn.com/image?fit=1280%2C720&url=https%3A%2F%2Fcdn0.tnwcndn.com%2Fwp-content%2Fblogs.dir%2F1%2Ffiles%2F2020%2F05%2Fcode-question-dev-gg.png&signature=2a6b283fed30239e7c62ddadcee0888b> -> The first image on the webpage.
- <https://www.economist.com/sites/default/files/images/2015/09/blogs/economist-explains/code2.png> -> The second image on the webpage.
- <https://s3.us-east-1.amazonaws.com/s3.discoveryplace.org/craft3/card-images/binary-code.jpg> -> The third image on the webpage.
- <https://www.computerhope.com/jargon/b/binary.htm> -> For the diagram that shows the conversion of decimals to binary.
- <https://medium.com/@savas/why-do-we-use-hexadecimal-d6d80b56f026> -> For the diagram that shows the conversion rates of decimal, binary and hex.
- Any content that was directly put on the website is linked and shown in brackets after the content that they represented on the website.
- Any sources that have been used as inspiration or used when a statistic has been mentioned during the report has been put in brackets after their respective section on the report.

P.S. If the pop-up tab isn't working, it is because of the Same Origin Policy on browsers that prevents it from acquiring the .htm files from the folder that everything is contained in. It would work if the link was a http or https or if the website is opened using the IIS (Internet Information Services). I used the IIS when developing the website and it was working perfectly then.